

IAL / IFJ

Interpret imperatívneho jazyka IFJ14

Tým 2
Variant a/1/II

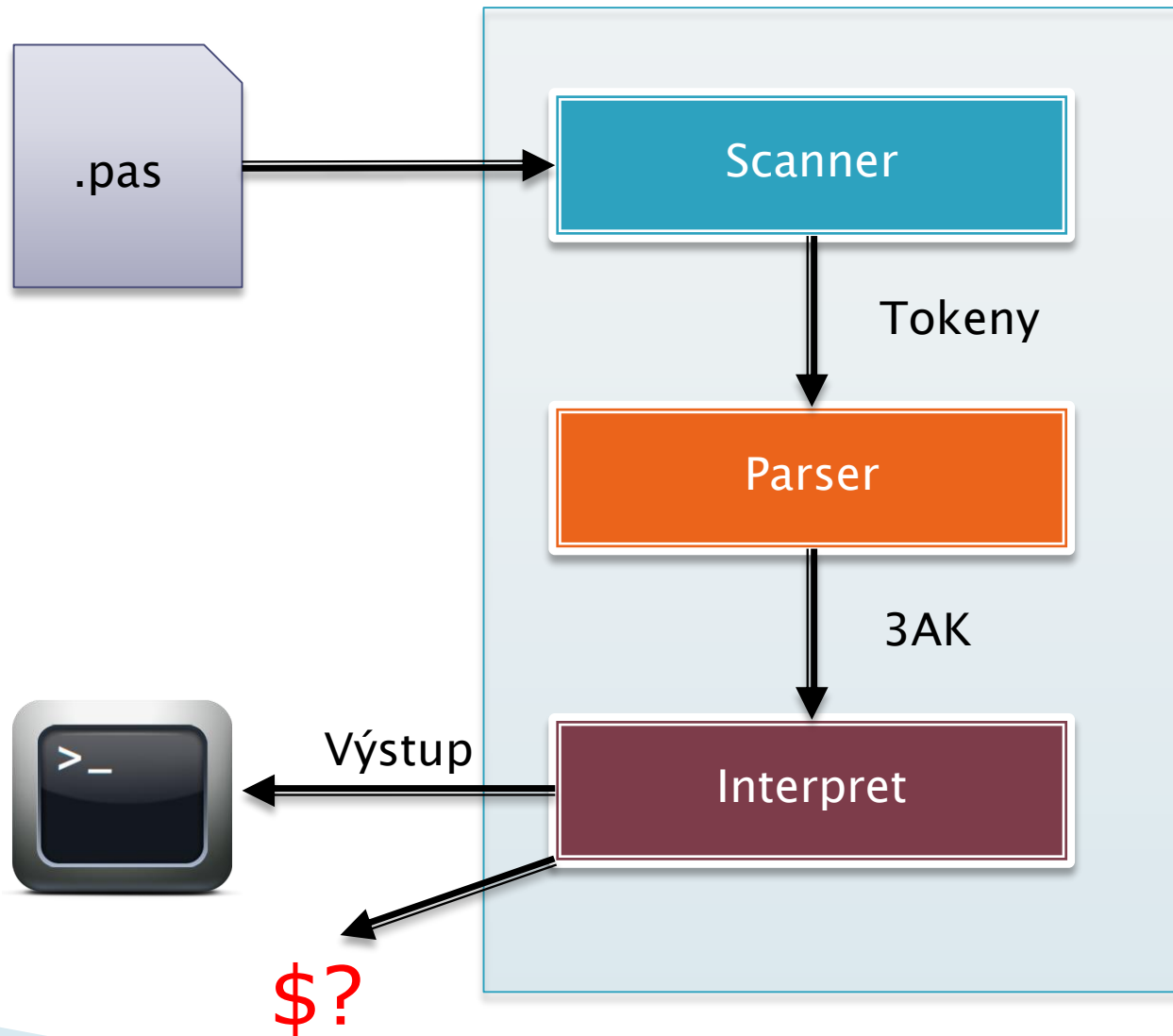
Členovia tímu

- ▶ Dávid Mikuš
- ▶ Peter Hostačný
- ▶ Tomáš Kello
- ▶ Adam Lučanský
- ▶ Michaela Lukášová

Obsah

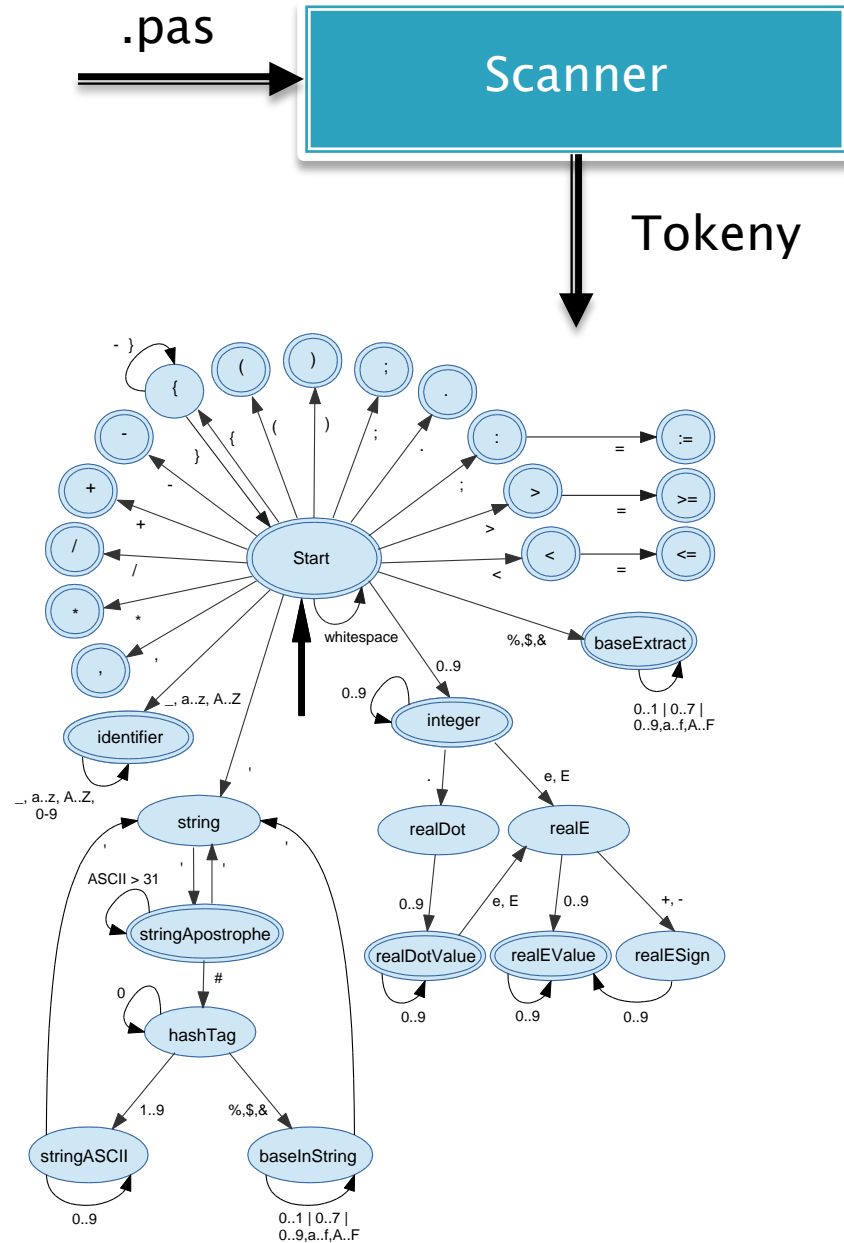
- ▶ Štruktúra interpretu
 - Scanner
 - Parser
 - Interpret
- ▶ Algoritmy

Štruktúra interpretu



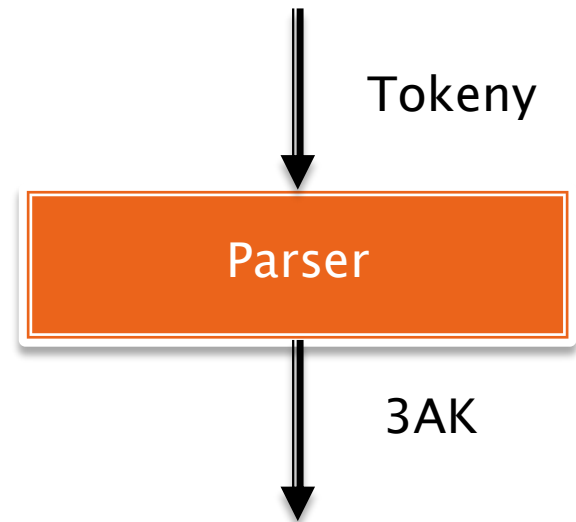
Scanner (BASE)

- ▶ Realizované pomocou konečného automatu
- ▶ Vektor tokenov
- ▶ Konverzia konštánt
- ▶ Detekcia kľúčových slov



Parser (MINUS, FUNEXP, BOOLOP, ELSEIF, REPEAT)

- ▶ Syntaktická analýza pomocou rekurzívneho zostupu
- ▶ Vyhodnocovanie výrazov (precedenčná analýza)
- ▶ Sémantické kontroly
- ▶ Optimalizácie (redukcia konštánt, inštrukcií)
- ▶ Generovanie trojadresného kódu



Interpret

- ▶ Inštrukcie sú inšpirované architektúrou x86 (zásobník, inštrukcie)
- ▶ Neobsahuje ukazovateľ na vrchol zásobníka (SP)
- ▶ Beží v nekonečnom cykle (HALT)

```
=====
TAPE:                                     offset
0      Instr_HALT                         0
1      Instr_PUSH_C                       0
2      Instr_PUSH_C                       1
3      Instr_PUSH_C                       2
4      Instr_PUSH_C                       3
5      Instr_MOV_GI                       1
6      Instr_L_GCII                       3
7      Instr_JMP_F                        10
8      Instr_PUSH_C                       3
9      Instr_WRITE                        1
10     Instr_JMP                          22
11     Instr_PUSH_C                       3
12     Instr_MOV_GI                       2
13     Instr_G_GCII                       3
14     Instr_JMP_F                        19
15     Instr_MUL_GGII                     3
16     Instr_MOV_GI                       2
17     Instr_SUB_GCII                     3
18     Instr_MOV_GI                       1
19     Instr_JMP                          12
20     Instr_PUSHX_GI                     3
21     Instr_PUSH_C                       4
22     Instr_WRITE                        2
23     Instr_HALT                         2
TAPE DUMP OVER === IP: 1 ==
```

3AK

Interpret

Generovanie aritmetických inštrukcií

·
·
·

	SRC 1	SRC 2	SRC 1	SRC 2	
	OP	lokalita1	lokalita2	typ1	typ2

MUL	local	local	int	int
2	0	0	2	2

$0010\ 00\ 00\ 10\ 10_{(2)} = 522$

← Výber z poľa ukazovateľov na funkcie

JMP

PUSH_C

MOV_GI

G_GCII

JMP_F

MUL_LLII

MOV_GI

Inštrukčná páska

Generovanie aritmetických inštrukcií

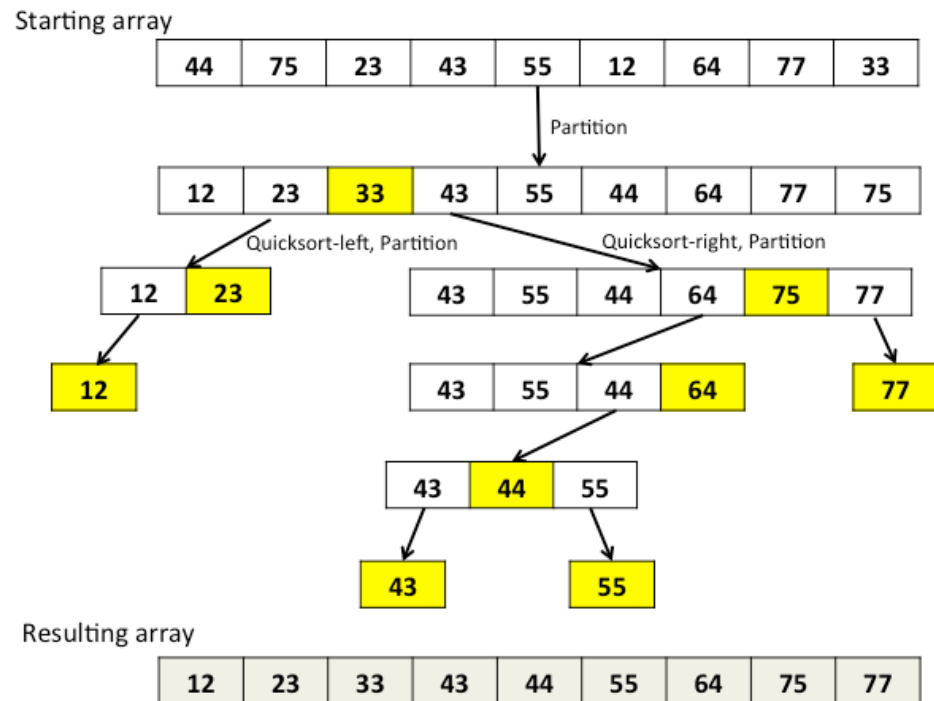
522 = $0010\ 00\ 00\ 10\ 10_{(2)}$

518	Instr_MUL_LLDI	4028d0
...		
521	Instr_MUL_LLID	4027e0
522	Instr_MUL_LLII	402700
...		
549	Instr_MUL_LCDD	402650

```
0000000000402700 <Instr_MUL_LLII>:
402700: 50                push    rax
402701: 49 89 f8          mov     r8,rdi
402704: 64 48 8b 04 25 28 00 mov     rax,QWORD PTR fs:0x28
40270b: 00 00
40270d: 48 89 04 24       mov     QWORD PTR [rsp],rax
402711: 48 8b 3d 38 ea 22 00 mov     rdi,QWORD PTR [rip+0x22ea38]
402718: 48 8b 4f 10       mov     rcx,QWORD PTR [rdi+0x10]
40271c: 48 8b 35 25 ea 22 00 mov     rsi,QWORD PTR [rip+0x22ea25]
402723: 49 8b 50 20       mov     rdx,QWORD PTR [r8+0x20]
402727: 48 01 f2          add     rdx,rsi
40272a: 48 c1 e2 04       shl     rdx,0x4
```

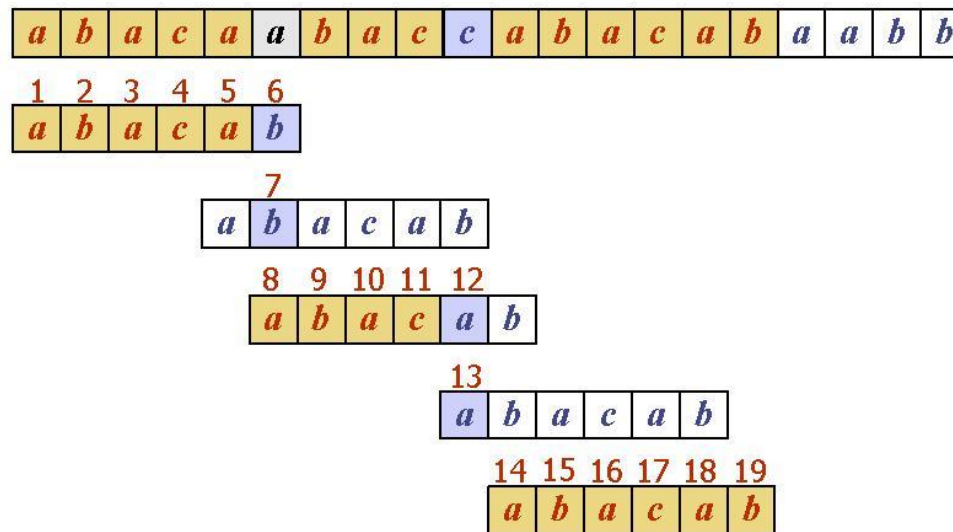
Algoritmy – Quicksort

- ▶ Algoritmus radenia.
- ▶ Nestabilný
- ▶ Neprirodzený
- ▶ $O(n * \log n)$
- ▶ Popr. $O(n^2)$



Algoritmy – Knuth–Morris–Pratt

- ▶ Vyhľadávanie podreťazca.
- ▶ Časová zložitosť: $O(n)$

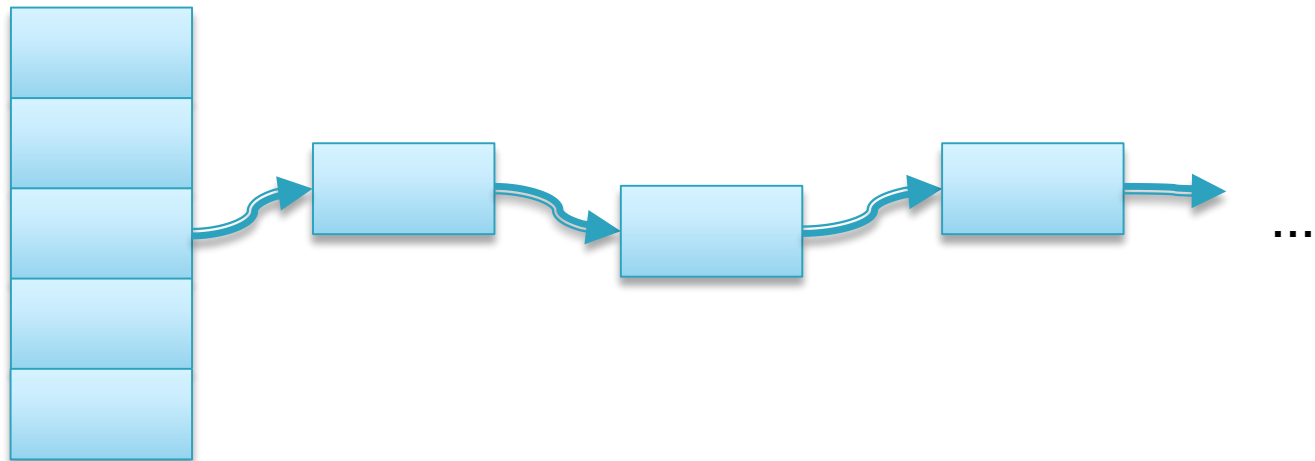


x	0	1	2	3	4	5
$P[x]$	<i>a</i>	<i>b</i>	<i>a</i>	<i>c</i>	<i>a</i>	<i>b</i>
$f(x)$	0	0	1	0	1	2

Algoritmy

Tabuľka s rozptýlenými polom

- ▶ Hashovacia funkcia SDBM
- ▶ Realizované pomocou poľa ukazovateľov na jednosmerne viazané zoznamy



Ďakujeme za pozornosť